

# 仮想計算機における FreeBSD の ファイルシステムのアウトソーシング

水野 貴史<sup>†</sup> 豊岡 拓<sup>††</sup> 小沢 健史<sup>††</sup>  
新城 靖<sup>††</sup> 佐藤 聡<sup>††</sup>  
中井 央<sup>†††</sup> 板野 肯三<sup>††</sup>

## 1. はじめに

仮想計算機技術の一つである仮想計算機モニタ (Virtual Machine Monitor, VMM) は、仮想的に計算機、またはその一部をエミュレートするソフトウェアである。VMM には Type I と Type II の二種類があり、Type I は実機上で VMM が直接動作するのに対し、Type II は実機上で動作する OS 上で、ユーザプログラムとして動作する。Type II の仮想計算機モニタにおいて、仮想計算機内で動作する OS をゲスト OS、実機上で動作する OS をホスト OS と呼ぶ。

Type II 仮想計算機には、入出力が遅いという問題がある。この問題を改善する手法の一つとして、準仮想化と呼ばれるものがある。これは、ゲスト OS を VMM 上で動くことを前提としてゲスト OS に手を加えるものである。準仮想化では、ゲスト OS に専用のデバイスドライバを組み込みことにより、ゲスト OS のディスク I/O を始めとした I/O 要求を VMM を通してホスト OS に送っている。

Type II の仮想計算機の入出力を高速化するため、本研究ではアウトソーシングという手法を提案している。アウトソーシングでは、デバイスドライバではなく、ソケットや VFS などのより高いレベルで抽象化を行う<sup>1)2)</sup>。この手法では、ゲスト OS の高水準な I/O 要求をホスト OS に伝えることで、I/O の高速化を行っている。

アウトソーシングはソケットについて、ホスト OS が Linux、ゲスト OS が NetBSD, Linux および Windows である環境に適用され、この手法の有効性が確認されている。ファイルシステムについては、ホスト OS とゲスト OS が共に Linux である環境に適用され、この手法の有効性が確認されている。本研究では、ゲ

スト OS を FreeBSD、ホスト OS を Linux という異なる OS 間において、ファイルシステムのアウトソーシングが適用可能かどうかを検証する。

## 2. Linux における VFS のアウトソーシング

Linux のファイルアクセスでは、プロセスがファイルアクセスする際、システムコールを介して後述する仮想ファイルシステム (Virtual File System, VFS) 関数の呼び出しを行う。

VFS 層は、ファイルシステムの抽象化を行い、複数のファイルシステムの実装の違いを隠蔽している。例えば、ローカルファイルシステムである ext3 とネットワークファイルシステムである NFS において、ファイル操作する関数は異なっている。しかし、それぞれのファイルシステムが操作関数へのポインタの表を上位層に提供することで、同一のインタフェースでの透過的なファイルアクセスを実現している。なお、この表はファイルシステムごとに存在し、この表の関数群を VFS インタフェースと呼ぶことにする。

Linux におけるファイルシステムのアウトソーシング<sup>2)</sup>では、この VFS 層を利用している。ゲスト OS には NFS のクライアントにあたるゲストモジュールを組み込み、ホスト OS には NFS のサーバにあたるホストモジュールを組み込む。ゲストモジュールは VFS インタフェースを持っており、VFS からの処理要求を VM RPC<sup>3)</sup> による通信を介してホスト OS のホストモジュールに伝える。ホストモジュールでは、送られてきた要求をホスト OS のファイルシステムにて処理を行い、その結果を返す。Linux におけるファイルシステムのアウトソーシングでは、このようにしてゲスト OS のファイルシステムやデバイスドライバを省き、ファイルアクセスの高速化を実現している。

<sup>†</sup> 筑波大学第三学群情報学類  
<sup>††</sup> 筑波大学システム情報工学研究科  
<sup>†††</sup> 筑波大学図書館情報メディア研究科

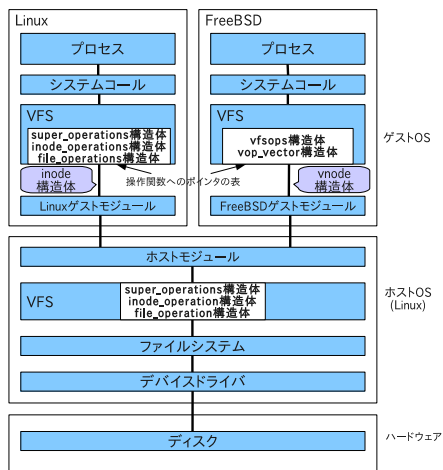


図 1 Linux, および FreeBSD におけるファイルシステムのアウトソーシング

### 3. FreeBSD における VFS のアウトソーシング

FreeBSD のファイルアクセスについても、Linux と同様に VFS 層による抽象化が行われているが、具体的な実装は大きく異なっている。ファイルシステムの操作関数へのポインタの表を、Linux では inode\_operations 構造体と file\_operations 構造体、および super\_operations 構造体として持っており、FreeBSD では vop\_vector 構造体と vfsops 構造体として持っている (図 1)。

ホスト OS とゲスト OS が共に Linux の場合、VFS インタフェースは同一なので、比較的容易にアウトソーシングを実現できた。しかし、FreeBSD ゲストの VFS 層からの処理要求が、現在 Linux で動いているホストモジュールに適合させられるか不明である。

そこで、本研究ではアウトソーシングにおいて FreeBSD と Linux の VFS インタフェースの違いをゲストモジュールで吸収する。ゲストモジュール内で引数の数や型を書き換えた後に、VM RPC を介してホスト OS のホストモジュールへと伝える。また、本研究の結果 Linux 側のホストモジュールのインタフェースに不都合が見つかった場合には、ホストモジュールのインタフェースの変更も行う。これにより、FreeBSD と Linux という異なる OS 間でファイルシステムのアウトソーシングを実現する。

本研究では、ゲストモジュールの実装には PUFFS<sup>4)</sup> を用いる。PUFFS とは、ユーザ空間にてファイルシステムの実装、および実行を可能としたフレームワークである。PUFFS で開発したファイルシステムは FUSE とは異なり、そのままカーネル空間にて動作させることも可能である。

本研究では Linux KVM<sup>5)</sup> において VM RPC を FreeBSD ゲストで利用可能にする。VM RPC とは、ゲスト OS からホスト OS へと RPC を行う仕組みである。VM RPC ゲストモジュールのポーティングは、VM RPC ゲストモジュールの OS への依存度が低いいため、低コストで行える。VM RPC ゲストモジュールが OS に依存している箇所は、カーネルへモジュールをロードする箇所、イベントキューのメモリ確保、およびヘッダの 3 箇所である。

### 4. 関連研究

TypeII VMM である VMware で用いられている HGFS<sup>6)</sup> も、抽象度の高い層で処理をホスト OS に投げるアプローチを採用している。HGFS ではホストモジュールがユーザ空間で実行されているのに対して、本研究ではカーネル空間で実現されている点が異なっている。

### 5. おわりに

現在までに、ゲスト OS として利用する FreeBSD のファイルアクセス、カーネル内部でのシステムコールの扱いなどについての調査と、VM RPC ゲストモジュールの FreeBSD への移植を行った。今後は PUFFS を使い、ユーザ空間でのファイルシステムの実装を進めていく。そして、ユーザ空間での実装が完了した後は、カーネル空間で動作することを確認する。また、アウトソーシングを FreeBSD をゲスト OS としたソケットなどにも適用し、その有効性の確認を進めていく。

### 参考文献

- 1) Eiraku, H., Shinjo, Y., Pu, C., Kohand, Y. and Kato, K.: Fast Networking with Socket-Outsourcing in Hosted Virtual Machine Environments, *ACM Symposium on Applied Computing (SAC2009)* (2009).
- 2) 豊岡拓, 新城靖, 齊藤剛: ホスト型仮想計算機環境におけるファイル入出力のアウトソーシングによる高速化, 第 21 回 コンピュータシステム・シンポジウム (ComSys 2009) (2009).
- 3) 齊藤剛, 新城靖, 榮樂英樹, 佐藤聡, 中井央, 板野肯三: 仮想計算機におけるアウトソーシングのためのゲスト-ホスト間 RPC, 第 20 回コンピュータシステム・シンポジウム (ComSys 2008) ポスター・デモセッション (2008).
- 4) Kantee, A.: puffs - Pass-to-Userspace Framework File System, *AsiaBSDCon 2007* (2007).
- 5) Kivity, A.: KVM : the Linux virtual machine monitor, *OLS '07: The 2007 Ottawa Linux Symposium*, pp. 225-230 (2007).
- 6) VMware, Inc.: Workstation User's Manual, [http://www.vmware.com/pdf/ws65\\_manual.pdf](http://www.vmware.com/pdf/ws65_manual.pdf).